

Инструменты обнаружения

Как загружать в Р-Сервис данные из инструментов обнаружения ИТ-инфраструктуры



СОДЕРЖАНИЕ

Введение	3
1 Интеграция с инструментами обнаружения	4
2 Использование GraphQL.....	5
3 Использование API импорта	7



ВВЕДЕНИЕ

Один из распространенных способов использования API¹ импорта – создание интеграции с одним или несколькими инструментами обнаружения. Например, инструменты Microsoft System Center Configuration Manager (SCCM)² или HP Network Node Manager (NNM)³ могут обнаруживать аппаратные и программные единицы в сети. Как правило, такие инструменты также способны находить некоторые связи между обнаруженными единицами. Эта информация может быть использована для автоматического добавления и обновления продуктов, конфигурационных единиц (КЕ) и связей между КЕ в Р-Сервис.

В качестве альтернативы использованию API импорта мы также предлагаем параметр-мутацию «discoveredConfigurationItems» (обнаружение конфигурационных единиц) в нашем API GraphQL⁴. Преимущество этого API заключается в том, что информация о наборе конфигурационных единиц (и их категориях продуктов, продуктах и связях) может быть отправлена за один вызов.

¹ API (Application Programming Interface) – обозначает «программный интерфейс приложения». Это набор правил, протоколов и инструментов, позволяющий одной компьютерной программе взаимодействовать с другой, обмениваясь данными и функциями.

² Microsoft System Center Configuration Manager (SCCM) – название инструмента для централизованного управления большим количеством компьютеров и устройств.

³ HP Network Node Manager (NNM) – название системы для мониторинга, анализа и управления ИТ-инфраструктурой.

⁴ API GraphQL – язык запросов и серверная среда для API с открытым исходным кодом. Позволяет клиенту очень точно описывать, какие именно данные ему нужны, а сервер отдаёт только эти данные.



1 ИНТЕГРАЦИЯ С ИНСТРУМЕНТАМИ ОБНАРУЖЕНИЯ

С системой Р-Сервис можно интегрировать несколько инструментов обнаружения. Чтобы Р-Сервис мог различать разные инструменты обнаружения, для каждого из них следует выбрать уникальный идентификатор. Идентификатор инструмента обнаружения затем можно сохранить в поле Источник (Source) в записях Р-Сервис, которые генерируются и поддерживаются этим инструментом. Примеры таких идентификаторов: «SCCM» или «NNM». Для заполнения поля Источник в записи также можно использовать более длинные имена с пробелами (максимум до 30 символов).

Кроме того, каждой конфигурационной единице (KE) и продукту следует присвоить уникальный идентификатор источника (Source ID), чтобы определять его независимо от идентификатора, присвоенного системой Р-Сервис. Использование поля Источник и поля Идентификатор источника важно как для импорта, так и для API GraphQL.



2 ИСПОЛЬЗОВАНИЕ GRAPHQL

Мы считаем, что использование GraphQL – это более простая альтернатива использованию файлов импорта. Для загрузки информации из инструментов обнаружения можно использовать параметр-мутацию «discoveredConfigurationItems» (обнаружение конфигурационных единиц). Преимущество этого API заключается в том, что информация о наборе конфигурационных единиц (и их категориях продуктов, продуктах и связях) может быть отправлена за один вызов. Мутация предоставит немедленный ответ, содержащий «асинхронный запрос» (AsyncQuery⁵), который можно использовать для получения результатов обработки всей отправленной информации.

Предполагаемое использование заключается в том, что все обнаруженные конфигурационные единицы (KE) отправляются партиями (по 100 записей). Затем можно опрашивать значения resultUrls (ссылки на результаты), возвращаемые «асинхронными запросами» (AsyncQueries), для получения результатов обработки каждой партии (в формате JSON⁶).

Как и в случае с API импорта, использование значений Источника (Source) и Идентификатора источника (Source ID) важно для правильного определения единиц без опоры на идентификаторы, генерируемые системой P-Сервис.

При загрузке обнаруженных записей таким способом необходимо указать категорию продукта и информацию о продукте. Это позволяет создавать новые записи без предварительной проверки их существования и, возможно, необходимости создавать их отдельно. Категории продуктов определяются не по их идентификатору источника (хотя он может быть указан), а по их ссылке, которая представляет собой неизменяемый идентификатор, автоматически присваиваемый каждой категории на основе ее имени при создании. При использовании параметра «DiscoveredItemStrategy» (стратегия обнаружения элементов) вы можете указать, должны ли предоставленные поля перезаписывать существующую информацию о продукте (категории) в системе P-Сервис или использоваться только для создания новых записей, если они еще неизвестны.

Помимо сопоставления конфигурационных единиц (KE) на основе присвоенных им источника и идентификатора источника, этот API также попытается сопоставить KE на основе их серийного номера при первоначальной загрузке. Это означает, что обнаруженные KE сопоставляются с KE, уже присутствующими в системе P-Сервис при первой загрузке (т.е. если в системе P-Сервис не найдена запись на основе их идентификатора источника). Таким образом, если KE с предоставленным серийным номером была введена вручную до ее обнаружения (и загрузки), загруженная информация используется для обновления вручную введенной KE вместо создания новой. Значение идентификатора источника, загруженное вручную, также присваивается введенной KE, поэтому будущие обновления применяются к нужной записи.

Обработка продуктов аналогична обработке конфигурационных единиц (KE), только вместо серийного номера используется идентификатор продукта. Таким образом, помимо поиска продуктов на основе присвоенных им источника и

⁵ AsyncQuery – это запрос к API, который не возвращает результат моментально. Он предоставляет ссылку на результат запроса, срок действия которой истекает и которая будет доступна после завершения выполнения запроса.

⁶ JSON (JavaScript Object Notation) – текстовый формат для хранения и передачи структурированных данных, основанный на синтаксисе JavaScript, но независимый от него.



идентификатора источника, этот API также попытается сопоставить их на основе идентификатора продукта при первоначальной загрузке. Это означает, что обнаруженные продукты сопоставляются с продуктами, уже присутствующими в системе Р-Сервис при первой загрузке (т.е. если в системе Р-Сервис не найдена запись на основе их идентификатора источника). Поэтому если продукт с указанным идентификатором введен вручную до того, как будут обнаружены (и загружены) какие-либо КЕ на его основе, загруженные КЕ связываются с этим продуктом, а новые не создаются. Значение идентификатора источника, загруженное вручную, также присваивается введенному продукту, поэтому будущие загрузки также будут связаны с нужной записью.



3 ИСПОЛЬЗОВАНИЕ API ИМПОРТА

Помимо использования GraphQL, API импорта также позволяет загружать информацию из инструментов обнаружения с помощью файлов. При таком подходе необходимо убедиться в существовании соответствующей записи о продукте (описывающей аппаратную модель или программное приложение конфигурационной единицы (KE)) до регистрации новой записи KE.

Важно заполнить столбец Источник (Source) для каждой строки при подготовке файла импорта с использованием данных из инструмента обнаружения. Кроме того, для файла импорта KE в столбце Идентификатор источника (Source ID) должен быть установлен уникальный идентификатор для каждой KE, по которому KE известна в инструменте обнаружения. Хранение значений Источник и Идентификатор источника в системе P-Сервис позволяет инструменту обнаружения обновлять запись KE, не зная ее идентификатора.

Ниже приведён пример того, как столбцы Источник и Идентификатор источника в файле импорта продуктов могут быть заполнены данными из инструмента обнаружения. Такой файл следует импортировать до импорта KE, чтобы гарантировать наличие всех необходимых продуктов в системе P-Сервис.

Источник (Source)	Идентификатор источника (Source ID)	Имя (Name)	Бренд (Brand)	Модель (Model)	Категория (Category)
SCCM	Abobe Reader	Abobe Reader	Adobe		software/browser_viewer_application
SCCM	Microsoft Windows 7	Microsoft Windows 7	Microsoft		software/operating_system_software
SCCM	VMware ESXi	VMware ESXi	VMware		software/operating_system_software
SCCM	Dell Precision M4400	Dell Precision M4400 Laptop PC	Dell	Precision M4400	computer/laptop_pc
SCCM	IBM Power 795	IBM Power 795 Server	IBM	Power 795	computer/server

После импорта продуктов можно импортировать CSV⁷- или TSV⁸-файл с информацией о конфигурационных единицах (KE). Однако перед началом импорта KE используйте функцию опроса, чтобы убедиться, что импорт продуктов завершен и прошел успешно.

⁷ CSV – простой текстовый формат для хранения табличных данных, где каждая строка файла представляет одну запись, а поля в строке разделены символом-разделителем (обычно запятой или точкой с запятой).

⁸ TSV – простой текстовый формат для хранения и передачи табличных данных, где каждая строка файла представляет собой одну строку таблицы, а ячейки (значения) внутри нее разделены специальным символом табуляции.



Пример ниже демонстрирует, как поля Источник и Идентификатор источника можно использовать в файле импорта KE, чтобы гарантировать, что KE могут быть однозначно идентифицированы с использованием данных только из инструмента обнаружения.

Источник (Source)	Идентификатор источника (Source ID)	Продукт (Product)	Метка (Label)	Имя (Name)	Идентификатор системы (System ID)
SCCM	AbobeReader9.1.0	Adobe Reader		Abobe Reader 9.1.0	
SCCM	MicrosoftWindows7 ProSP3	Microsoft Windows 7		Microsoft Windows 7 Pro SP3	
SCCM	VMwareESXiBuild164009	VMware ESXi		VMware ESXi 4.0 Build 164009	
SCCM	01:23:45:67:89:ab	Dell Precision M4400 Laptop PC	CMP00052	Dell Precision M4400 Laptop PC	http://sccm.example.com?mac=0123456789ab
SCCM	5691602	IBM Power 795	CMP00207	IBM Power 795 Server	http://sccm.example.com?id=5691602

Обратите внимание, что имена без пробелов использовались для определения идентификатора источника для KE программного обеспечения. Это было сделано для демонстрации того, как можно уменьшить длину текстовой строки в поле Идентификатор источника, которая может содержать максимум 128 символов.

Примеры URL⁹-адресов для поля Идентификатор системы являются гипотетическими.

После импорта KE можно импортировать связи между ними. Снова используйте функцию опроса, чтобы убедиться, что импорт KE завершен и прошел успешно.

Ниже приведен пример файла импорта связей между KE, который связывает операционную систему и приложение с персональным компьютером, идентифицируемым по его MAC¹⁰-адресу.

⁹ URL (Uniform Resource Locator) – обозначает «унифицированный указатель ресурса». Адрес интернет-ресурса, который использует браузер, чтобы найти и открыть нужную страницу, файл или сервис.

¹⁰ MAC-адрес (Media Access Control address) – уникальный идентификатор сетевого интерфейса, присваиваемый каждому устройству, подключённому к сети, – это аппаратный (физический) идентификатор устройства – «кто» оно есть.



Источник (Source)	Источник KE (CI Source)	Идентификатор источника KE (CI Source ID)	Источник связи KE (Related CI Source)	Идентификатор источника связи KE (Related CI Source ID)	Тип связи (Relation Type)
SCCM	SCCM	01:23:45:67:89:ab	SCCM	AdobeReader9.1.0	Ребенок (child)
SCCM	SCCM	01:23:45:67:89:ab	SCCM	MicrosoftWindows7ProSP3	Ребенок (child)

Если для KE в файле импорта определена одна или несколько связей и в первом столбце для этих связей указан источник, то все существующие связи этой KE с тем же источником будут удалены во время импорта.

Важно: убедитесь, что все связи для KE сгруппированы вместе в файле импорта. В противном случае последняя связь, указанная для KE в файле импорта, приведет к удалению предыдущих связей, добавленных во время импорта для этой же KE.

Если источник связи не определен в файле импорта, то связь будет добавлена во время импорта без удаления каких-либо существующих связей.

